

Amendments to the Specification:

Please replace paragraph [0017] with the following amended paragraph:

[0017] Figure 1 shows a computer system 100 constructed in accordance with the preferred embodiment. Computer system 100 generally comprises a microprocessor or CPU 20 coupled to a main memory 26 and various other peripheral computer system components, through an integrated host bridge 22. The CPU 20 preferably couples to the host bridge logic 22 via a host bus 24, or the host bridge logic 22 may be integrated into the CPU 20. The CPU 20 preferably comprises an Itanium™ microprocessor manufactured by Intel Corporation. It should be understood, however, that computer system 100 could comprise other types and brands of microprocessors as well. For example, computer system 100 may comprise a Pentium III™ or Pentium IV™ microprocessor, or any microprocessor later developed, by Intel Corporation. The CPU ~~400~~20 may also comprise any microprocessor made by Advanced Micro Devices. Thus, the computer system may implement other bus configurations or bus bridges in addition to, or in place of, those shown in Figure 1. Moreover, computer system 100 could also comprise several microprocessors, as may be used in applications such as a server system.

Please delete paragraph [0019]:

~~**[0019]** The computer system 100 also preferably comprises a graphics controller or video driver card 30 that couples to the host bridge 22 via an Advanced Graphics Port ("AGP") bus 32, or other suitable type of bus. Alternatively, the video driver card may couple to the primary expansion bus 34 or one of the secondary expansion buses, for example, PCI bus 40. Graphics controller 30 further couples to a display device 32 which may comprise any suitable electronic display device upon which any image or text can be represented.~~

Please replace paragraph [0022] with the following amended paragraph:

[0022] Referring still to Figure 1, a firmware hub 42 couples to the ICH 36 by way of the LPC bus 38. The firmware hub 46-42 preferably comprises Read Only Memory (ROM) which contains software programs executable by the CPU 20. The software programs preferably comprise not only programs to implement Basic Input/Output System (BIOS) commands, but also include instructions executed during and just after power on self test (POST) procedures. These software programs perform various functions including verifying proper operation of various system components before control of the system is turned over to the operating system.

Please replace paragraph [0023] with the following amended paragraph:

[0023] In broad terms, ~~t~~The preferred embodiments of the present invention are directed to synchronizing access by software streams to BIOS routines. The preferred embodiments were developed in the context of software streams accessing shared variables through the use of BIOS routines, and thus the following description is related to the context of development; however, the description in this manner should not be construed as a limitation as to the scope of applicability of the concepts described.

Please replace paragraph [0024] with the following amended paragraph:

[0024] Figure 2 shows a flow diagram which exemplifies the preferred procedure for a software thread or stream calling BIOS routines. In particular, the process starts at step 50 and proceeds directly to the step of opening the BIOS group (step 52). For some BIOS routines in a computer system, no synchronization is required, and thus these BIOS routines need not be opened and closed as described in the preferred embodiments. While it would be possible to require software streams to open and close the entire BIOS before executing any BIOS routines, preferably only certain groups of routines, those groups that manage access to shared variables, are opened and closed as described in the preferred embodiment. Moreover, opening a BIOS group should

not necessarily be construed to mean that more than one BIOS routine should be grouped for opening and closing purposes. It is possible that even a single BIOS routine, where that BIOS routine requires synchronization between multiple software streams, may comprise a BIOS group. Thus, in step 52, the calling software program attempts to open the BIOS group. Preferably, the next step in the procedure is to examine the return value from the open procedure (step 52). More particularly, if the return value is a valid handle (step 54), then the process continues to step 56 where the calling software stream, now holding the valid handle, is allowed to call BIOS routines to perform the desired operation on the shared variables. If, however, the return value from step 52 is not a valid handle, indicating that the BIOS is already opened and owned by another software stream, then preferably the calling software stream pauses momentarily (step 5859) and again attempts to open the BIOS group (step 52).

Please replace paragraph [0026] with the following amended paragraph:

[0026] Consider for purposes of explanation a first software stream and a second software stream, the software streams either executed on the same microprocessor or on different microprocessors. Further consider that both software streams need to access and/or update shared variables maintained through the use of BIOS routines. Still referring to Figure 2, consider that the first stream is the first to attempt to open the particular BIOS group of interest. In such a case, the first software stream moves through steps 50, 52, 54 and 56 of Figure 2. Consider also that the second software stream is the second to attempt to access the BIOS routines. In this case, the second software stream progresses through steps 50, 52 and 54 of Figure 2, but because the particular BIOS group is already open and owned by the first software stream, the return value from the attempt to open the BIOS group is not a valid handle. Thus, the second software stream begins a process of circularly attempting to open the BIOS group (step 52), examining the return value for indications of a proper handle (step 54), pausing (step 5859), and attempting again to open the BIOS group (again step 52).

Please replace paragraph [0030] with the following amended paragraph:

[0030] Although the steps described above may be implemented in any computer system, in the preferred embodiment, the steps are implemented in a system having at least one Itanium™ microprocessor made by Intel Corporation. ~~As one of ordinary skill in the art is aware, in~~ systems operating with an Itanium™ microprocessor and related chipset, BIOS routine calls are made to a System Abstraction Layer (SAL). For more information regarding the Itanium™ processor family system extraction layer, reference may be had to Intel document No. 245359-003 titled "Itanium™ Processor Family System Abstraction Layer Specification," dated January 2001, incorporated herein by reference as if reproduced in full below. Thus, rather than the traditional loading of a services number into a register of the microprocessor and issuing a software interrupt, in the Itanium™ system kernel mode software preferably communicates in a C language format function call with the system abstraction layer to request BIOS type services. While certain BIOS routines are generic to every computer system, OEMs have the ability to specify and use custom BIOS routines. Thus, additional routines may be added to the system abstraction layer of the Itanium™ processor family. In the preferred embodiments, implementing the open, use and close technique for routines that modify shared variables preferably takes place at the SAL level of the computer in a C language format function call. However, the same procedures and steps may be implemented directly at the BIOS routine level without departing from the scope and spirit of this invention.